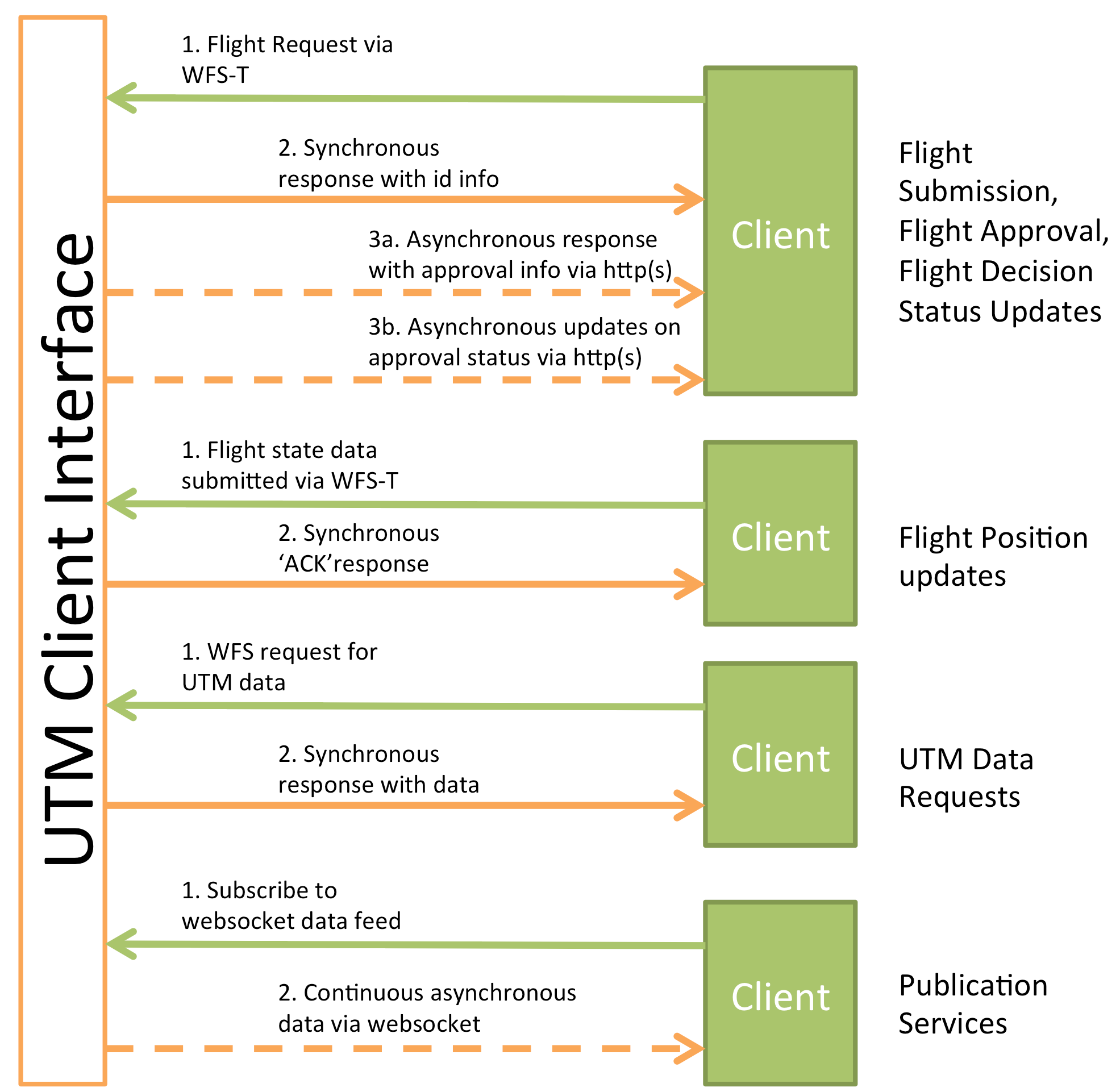
**UTM Client Interface Document Summary**

Draft document defining the interface between UTM system and clients.

**Client Functions**

* Flight Submission
  + Provide planned flights
  + Request to access airspace
* Flight Approval
  + Initial decision asap (asynchronous)
* Flight Decision Status Update
  + Updates to flight approval decision
  + Due to weather, outages, etc
* Flight Position Updates
  + Clients submit state info
  + Location @ time, heading, altitude
  + Future: Obtained through additional entity or collaboration between operators(?) and UTM maintainers
* UTM Data Requests
  + UTM has access to all flight plans
  + Provide complete picture, “common situational awareness for stakeholders”
* Publication Services
  + specific or general data feeds
    - new flight plans
    - constraints
    - general info
  + publish-subscribe

**Overall Architecture**



**Web Feature Service (WFS) request**

Mechanism to retrieve specific data from UTM

Standard for exchanging geospatial data

Defined in Open Geospatial Consortium (OGC)

**Details on WFS Communication**

“GetCapabilities”

Return XML document with details on what services can be provided

“FeaureTypeList” and “FeatureType” elements

Contained in XML document. List of features that can be retrieved from the UTM.

“DescribeFeatureType”

Returns specific info on feature

\*\*additional details provided on requesting information\*\*

In summary:

* WFS used for retreiving data relevant to client’s needs
  + Ex: All operations @ time and location
  + Ex: Mapping application requesting data for new area when view changes

WFS-T (Web Feature Service - Transaction)

**Flight Plan Submission and Approval/Denial (WFS-T)**

Submission of flight data is a “request”, does not immediately enter UTM

UTM checks validity, safety, constraints, etc

UTM decides if submission will be approved/denied

Operator provides callback URL for UTM to post decision on the flight back to the client. Client responsible for servicing URL.

JSON – UTM Decision Message Schema

\*\*see example in document\*\*

**Flight State Updates (WFS-T)**

Client submits info to a position table in database

Reference flight operation table via the operation id supplied in the original flight request submission

UTM gives success acknowledgement of flight update submission

**Publication Services**

Data about given area, operator, cancelled flight plans, etc

Ex. Track flight

Potentially (hopefully) Client-defined data streams

\*\* example shown on document p12\*\*

**Client Communication**

Clients request information

UTM handles authorization

SockJS, socket

**Authentication and Authorization**

Layer-level security

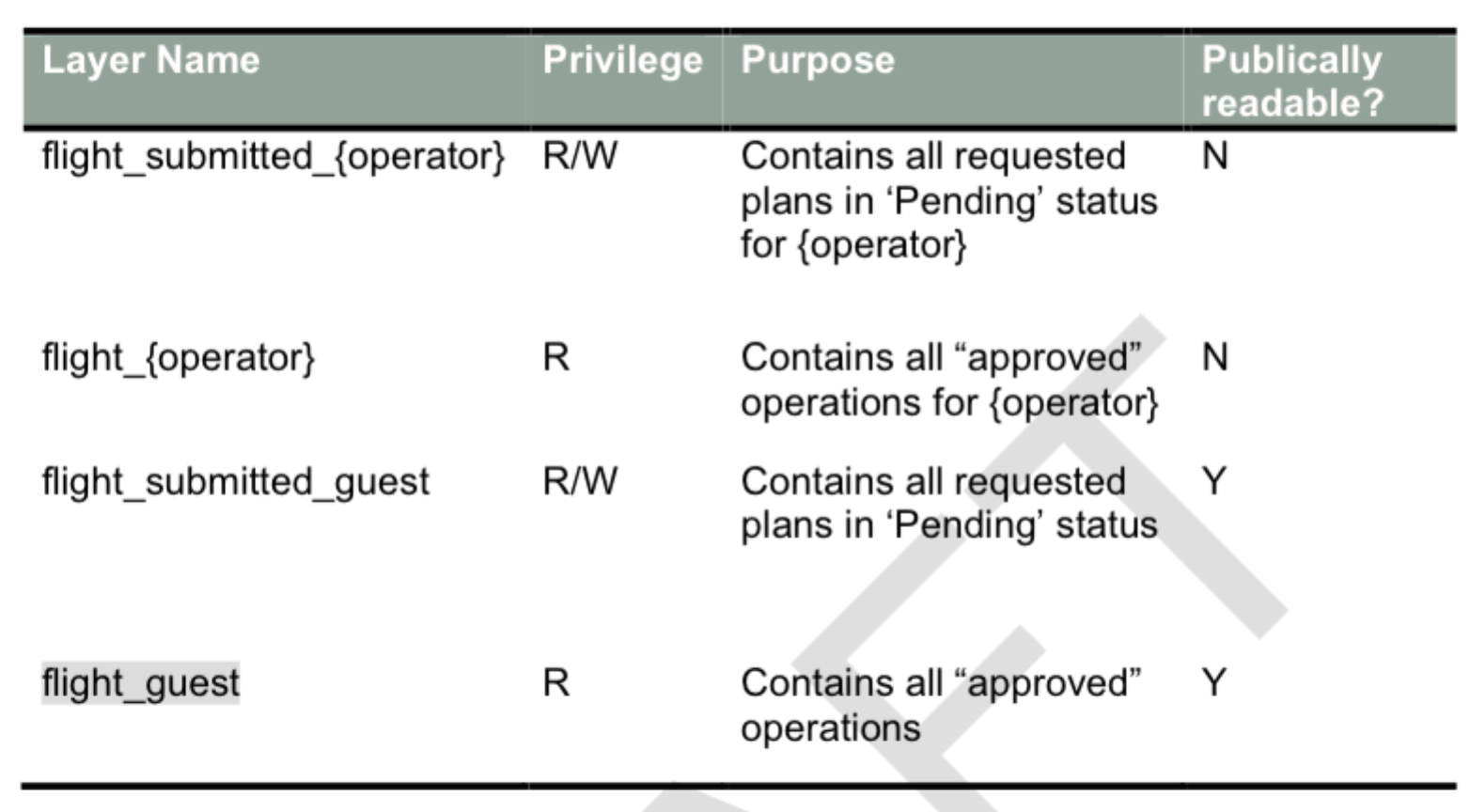
UTM obtains credentials from Lightweight Directory Access Protocol (LDAP)

Operator can only read/write data only to authorized layers

Future goal: Make all data in UTM publically readable to enhance common situational awareness

Each operator assigned a string to identify

Names are open to all, can create non-identifying string to remain anonymous



Credentials provided to each operator/user (username, password, operator ID)